

REMARKS

I. Introduction

Claims 1-32 are pending in the present application. In a July 21, 2006 Office Action (herein "Office Action"), Claim 25 was rejected under 35 U.S.C. § 101 for being directed to non-statutory subject matter. Claim 31 was rejected under 35 U.S.C. § 112 for an insufficient antecedent basis for a limitation in the claim. Claims 1, 2, 4-6, 8-15, 17-19, 21-30, and 32 are rejected 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 7,028,056, to Hendel et al. (herein "Hendel").

Applicants respectfully request reconsideration and allowance of the pending claims. Hendel fails to teach a system method or interface for consistently capturing kernel-resident information through the use of a requestor application and a kernel mode module. In addition to presenting the reasons why applicants believe that the pending claims are in condition for allowance, a brief summary of the present application, as well as the cited reference, Hendel, are presented. However, it should be appreciated that the brief summaries are presented solely to assist the Examiner in recognizing the differences between the pending claims and the cited reference, and should not be construed as limiting upon the present invention.

II. Summary of the Present Invention

The present application is generally related to a system, method, and interface for consistently capturing kernel-resident information. A user mode module initiates a kernel mode information request through an application program interface identifying one or more executing process threads. A kernel mode module captures information corresponding to standard kernel mode information and corresponding to the specifically identified process threads. The information is returned in a pre-allocated buffer.

In one example of the present invention, a requestor application transmits a request to collect kernel mode module information. The request to collect kernel mode module information

includes an identification of one or more executing process threads from which kernel mode information will be collected. A kernel mode module obtains the request to collect kernel mode module information and captures information corresponding to each thread identified in the request to collect kernel mode module information. The kernel mode module transmits a result of the capturing of the information corresponding to each thread identified in the request to collect kernel mode module information. The requestor module obtains the result of the capturing of the information corresponding to each thread identified in the request to collect kernel mode module information.

III. U.S. Patent No. 7,028,056 to Hendel et al.

Hendel is purportedly directed toward methods and arrangements to reduce the amount of data required to conduct postmortem analysis following an operating system or application failure. See Hendel, Col. 2, lines 23-26. As taught in Hendel, this is accomplished through determining when to generate a dump file and subsequently generating said dump file to a storage medium. Col. 2, lines 41-46. Further, Hendel teaches that the dump file can be stored to a storage medium and accessed for subsequent analysis. Col. 2, lines 47-48. Further, Hendel teaches that a user minidump file can be generated containing the list of threads running at the time of the crash; the thread-context and callstack for each running thread; the list of modules loaded at the time of the crash; the reason for the crash; and, selected regions of the process's memory that relate to the cause of the crash. Col. 3, lines 4-10. Further, Hendel teaches functions that allow an application to write and read a user minidump file. Col. 8, lines 37-41. Further, Hendel teaches a WriteMiniDump API which reads a user-mode minidump to a file. Col. 9, lines 14-16. Still further, Hendel teaches a ReadMiniDump API reading a field from a user-mode minidump. Col. 9, lines 54-56. Even further, the kernel minidump file of Hendel is accessible only after a reboot. See Figure 2, col. 7 lines 15-21.

Hendel fails to teach or suggest a system, method, and interface for capturing kernel-resident information that includes an identification of one or more executing process threads. Further, Hendel does not teach a requestor application requesting or receiving information directly from a kernel mode module. Still further, Hendel does not teach a kernel mode module receiving a request for kernel mode module information, capturing the requested information, and transmitting the captured information to the requestor application.

IV. Claim Rejections

A. 35 U.S.C. § 101 Rejection of Claim 25

Claim 25 was rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The Office Action states that, "The claim cites 'a software architecture' comprising components and API's, no hardware structure is present on which to store the software or execute the software." See Office Action, p. 2. In response, applicants have amended the claim to reflect the hardware structure. Applicants further request that the 35 U.S.C. § 101 rejection of Claim 25 be withdrawn.

B. 35 U.S.C. § 112 Rejection of Claim 31

Claim 31 was rejected under 35 U.S.C. § 112. In response, applicants have amended the claim to reference Claim 30 which provides the proper antecedent basis for "all pending I/O packets." Accordingly, applicant requests that the 35 U.S.C. § 112 rejection of Claim 31 be withdrawn.

C. 35 U.S.C. § 102(e) Rejections of Claims 1, 2, 4-6, 8-15, 17-19, 21-30, and 32

1. Introduction

Claims 1, 2, 4-6, 8-15, 17-19, 21-30, and 32 were rejected under 35 U.S.C. § 102(e) as being anticipated by Hendel. For the following reasons, applicants respectfully submit that the pending claims application are not anticipated by Hendel.

a. Independent Claim 1

With regard to Claim 1, applicants submit that Hendel fails to teach each and every element of Claim 1. Claim 1 recites:

1. In a computer system having an operating environment including user mode modules having a first level of protection and kernel mode modules having a second level of protection, a method for consistently collecting information associated with the execution of a user mode module, the method comprising:

transmitting, by a requestor application, a request to collect kernel mode module information, wherein the request to collect kernel mode module information includes an identification of one or more executing process threads from which kernel mode information will be collected;

obtaining, by a kernel mode module, the request to collect kernel mode module information;

capturing, by the kernel mode module, information corresponding to each thread identified in the request to collect kernel mode module information;

transmitting, by the kernel mode module, a result of the capturing of the information corresponding to each thread identified in the request to collect kernel mode module information; and

receiving, by the requestor application, the result of the capturing of the information corresponding to each thread identified in the request to collect kernel mode module information.

Claim 1 is directed toward a method by which a requestor application transmits a request to collect kernel mode module information from a kernel mode module. The kernel mode module receives the request and then captures information corresponding to the identified threads, and transmits it back to the requestor application. The requestor application obtains the result of the capturing of the information corresponding to each thread identified in the request to collect kernel mode module information. The process occurs without requiring a crash. The user is not required to reboot his machine to access the information received by the requestor application.

In contrast to the claims of the present application, Hendel is directed toward a method of analyzing information following an operating system or application failure. Hendel teaches generating a dump or "minidump" file by gathering "the thread, thread context, and callstack for the thread that caused the failure," "the process containing the failing thread," and "the reason for the crash." Col. 2, lincs 41-53. Further, Hendel teaches "the resulting dump file can then be stored to a storage medium and accessed for subsequent analysis." Col. 2, lines 47-49. Thus, in accordance with the teachings of Hendel, a variety of information is being gathered following a failure or crash and then placed within a dump file. This file can subsequently be accessed to conduct analysis. The kernel minidump file of Hendel is accessible only after a reboot. See Figure 2, col. 7 lines 15-21.

Because Hendel is limited to post-crash error analysis, Hendel clearly does not teach "an identification of one or more executing process threads" and "capturing by kernel mode module, information corresponding to each thread identified in the request to collect kernel mode module information" as recited in Claim 1. To anticipate a claim under 35 U.S.C. § 102(e), the cited reference must teach each and every element recited in the claim. *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631 (Fed. Cir. 1987). With regard to Claim 1, applicants respectfully submit that the cited reference, Hendel, fails to teach at least a system, method, and interface for consistently capturing kernel-resident information without requiring an intrusive or destructive collection mechanism such as a system crash or generation of a dump file. Additionally, applicants respectfully submit that the cited reference fails to teach "an identification of one or more executing process threads" and "capturing by kernel mode module, information corresponding to each thread identified in the request to collect kernel mode module information" as recited in Claim 1. For these and the aforementioned reason, applicants respectfully request a withdrawal of the 35 U.S.C. § 102(e) rejection with regard to Claim 1.

b. Dependent Claims 2, 4-6

Claims 2 and 4-6 are dependent on Claim 1. As discussed above, Hendel fails to teach or suggest all the limitations recited with regard to Claim 1. Accordingly, for the abovementioned reasons, Claims 2 and 4-6 are allowable over the cited art. In addition, Claims 2 and 4-6 add to the patentability and nonobviousness of applicants' invention. For these reasons, applicants respectfully request withdrawal of the 35 U.S.C. § 102(e) rejections with regard to Claims 2 and 4-6 and allowance of the claims.

c. Dependent Claim 8

Claims 8 is dependent on Claim 1. As discussed above, Hendel fails to teach or suggest all the limitations recited with regard to Claim 1. Accordingly, for the above-mentioned reasons, Claims 8 is allowable over the cited art. In addition, Claim 8 adds to the patentability and nonobviousness of applicants' invention, the details of which are discussed below. Dependent Claim 8 adds to the patentability of applicants' invention "capturing information corresponding to each thread identified in a request to collect kernel mode module information is asynchronous." Thus, information regarding an executing thread may be requested at any time. Although Hendel teaches in an application is allowed to write a user-made dump file at any time, it is again limited to the dump file. Hendel clearly does not teach an application can capture information corresponding to a specific identified thread at any time as recited in Claim 8. For these reasons, applicants respectfully request withdrawal of the 35 U.S.C. § 102(e) rejections with regard to Claim 8 and allowance of the claims.

d. Dependent Claims 9-13

Claims 9-13 are dependent on Claim 1. As discussed above, Hendel fails to teach or suggest all the limitations recited with regard to Claim 1. Accordingly, for the abovementioned reasons, Claims 9-13 are allowable over the cited art. In addition, Claims 9-13 add to the

patentability and nonobviousness of applicants' invention. For these reasons, applicants respectfully request withdrawal of the 35 U.S.C. § 102(e) rejections with regard to Claims 9-13 and allowance of the claims.

e. Independent Claim 14

With regard to Claim 14, applicants submit that Hendel fails to teach each and every element of Claim 14. Claim 14 recites:

14. In a computer system having an operating environment including user mode modules having a first level of protection and kernel mode modules having a second level of protection, a method for consistently collecting information associated with the execution of a user mode module, the method comprising:

obtaining a user mode module request to collect kernel mode module information including an identification of one or more executing process threads from which kernel mode information will be collected;

capturing information corresponding to each thread identified in the request to collect kernel mode module information; and

transmitting the captured kernel mode module information.

As discussed above, Hendel is directed toward the method of analyzing information following an operating system or application failure. Hendel teaches generating a dump or minidump file for subsequent analysis. Col. 2, lines 47-49. The kernel minidump file of Hendel is accessible only after a reboot. See Figure 2, col. 7 lines 15-21. Thus, Hendel clearly does not teach "obtaining a user mode module request to collect kernel mode module information including an identification of one or more executing process threads from which kernel mode information will be collected" as recited in Claim 14. For those and the aforementioned reasons regarding Claim 14, applicants respectfully request the withdrawal of the 35 U.S.C. § 102(e) rejection with regard to Claim 14.

f. Dependent Claims 15, 17-19, and 21-24

Claims 15, 17-19, and 21-24 are dependent on Claim 14. As discussed above, Hendel fails to teach or suggest all the limitations recited with regard to Claim 14. Accordingly, for the abovementioned reasons, Claims 15, 17-19, and 21-24 are allowable over the cited art. In addition, Claims 15, 17-19, and 21-24 add to the patentability and nonobviousness of applicants' invention. For these reasons, applicants respectfully request withdrawal of the 35 U.S.C. § 102(e) rejections with regard to Claims 15, 17-19, and 21-24 and allowance of the claims.

g. Independent Claim 25

With regard to Claim 25, applicants submit that Hendel fails to teach each and element of Claim 25. As amended, Claim 25 recites:

25. In a computer system having a processor, a memory, and an operating environment, the operating environment including user mode modules having a first level of protection and kernel mode applications having a second level of protection, a software architecture for consistently collecting information associated with the execution of a user mode module, the architecture comprising:

a processing component for capturing kernel mode module information corresponding to one or more executing processing threads identified in a request to collect kernel mode module information; and

at least one application program interface for accessing the processing component and identifying the one or more executing processing threads from which to collect kernel mode module information.

As discussed above, Hendel teaches that their methods can be used to provide application programming interfaces (APIs) or like capability when writing and reading dump files. Col. 3, lines 21-26. Thus, Hendel clearly does not teach "at least one application program interface for accessing the processing component and identifying the one or more executing processing threads from which to collect kernel mode module information."

With regard to Claim 25, applicants respectfully submit that the cited reference, Hendel, fails to teach at least a software architecture for consistently collecting information associated with the execution of a user mode module. Additionally, applicants respectfully submit that the cited reference fails to teach "a processing component for capturing kernel mode module information corresponding to one or more executing processing threads identified in a request to collect kernel mode module information" and " at least one application program interface for accessing the processing component and identifying the one or more executing processing threads from which to collect kernel mode module information" as recited in Claim 25. For those and the aforementioned reasons regarding Claim 25, applicants respectfully request the withdrawal of the 35 U.S.C. § 102(e) rejection with regard to Claim 25.

h. Dependent Claims 26-30 and 32

Claims 26-30 and 32 are dependent on Claim 25. As discussed above, Hendel fails to teach or suggest all the limitations recited with regard to Claim 25. Accordingly, for the above-mentioned reasons, Claims 26-30 and 32 are allowable over the cited art. In addition, Claims 26-30 and 32 add to the patentability and nonobviousness of applicants' invention. For these reasons, applicants respectfully request withdrawal of the 35 U.S.C. § 102(e) rejections with regard to Claims 26-30 and 32 and allowance of the claims.

V. Conclusion

Based on the above-reference arguments, applicants respectfully submit that all the pending claims of the present application, Claims 1, 2, 4-6, 8-15, 17-19, 21-30, and 32 are allowable over the cited and applied references. Accordingly, applicants respectfully request withdrawal of all the rejections of the claims of the present invention and allowance of the

present application. If any questions remain, applicants request that the Examiner contact the undersigned at the telephone number listed below.

Respectfully submitted,

CHRISTENSEN O'CONNOR
JOHNSON KINDNESS^{PLLC}

A handwritten signature in dark ink, appearing to be 'M. Uribe', followed by a horizontal line and a circular flourish.

Mauricio A. Uribe
Registration No. 46,206
Direct Dial No. 206.695.1728

MAU/MLR:pwv

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100